

## Search Detail

Appl. 10/081079

Date	Type	Search
26 May 2004	R	stored procedure compile
	R	stored procedure and compile
	R	stored procedure

10/081079

	Hits	Search Text	DBs	Time Stamp ▽
1	22	5875334.URPN.	USPAT	2004/05/26 09:39
2	7	("5201046"   "5291582"   "5432930"   "5487132"   "5546570"   "5617567"   "5675804").PN.	USPAT	2004/05/26 09:37
3	236	(707/1-206).ccls. and (stored adj procedure\$1) and compil\$5	USPAT; US-PGPUB	2004/05/25 16:51
4	49	(707/1-206).ccls. and ((stored adj procedure\$1) same compil\$5)	USPAT; US-PGPUB	2004/05/25 16:51
5	513	(707/1-206).ccls. and (stored adj procedure\$1)	USPAT; US-PGPUB	2004/05/25 16:50
6	9	("5499368" "5546526" "5732274" "6212514" "6212514" "5819255" "6507834" "6405212" "5857195" "5706494").pn.	USPAT; US-PGPUB	2004/05/25 16:49
7	10	("6009265" "6098075" "5940289" "6564205" "6006224" "6073129" "6105033" "6446062" "6023696" "5930800").pn.	USPAT; US-PGPUB	2004/05/25 16:46
8	10	("5890148" "5410693" "5819253" "5903898" "4506326" "5960426" "6085189" "5210686" "5574900" "5873098").pn.	USPAT; US-PGPUB	2004/05/25 16:41
9	10	("6732084" "6236986" "6044216" "5761493" "5826077" "6286135" "5884299" "5809505" "5659728" "5696960").pn.	USPAT; US-PGPUB	2004/05/25 16:37
10	10	("6477540" "5864840" "6032143" "6324683" "6438538" "5875334" "5553234" "5689633" "5742810" "5794231").pn.	USPAT; US-PGPUB	2004/05/25 16:33
11	12	stegelmann.in.	USPAT; US-PGPUB	2004/05/25 16:13

Set	Items	Description
S1	25199	(LOWLEVEL OR LOW()LEVEL OR ASSEMBLY OR MACHINE) (N) (LANGUAG- E? OR CODE?)
S2	4693879	EXPRESSION? OR PROCEDURE? OR FORMULA? OR ALGORITHM?
S3	2857	PRECOMPILE? OR PRE()COMPILE?
S4	96027	RUNTIME? OR RUN()TIME?
S5	12670285	EMBED? OR INTEGRAL? OR WITHIN? OR INTERNAL? OR INSIDE? OR - INTEGRATE?
S6	562	(DATABASE OR DATA() (BANK? OR BASE?) OR DATABANK? OR DB OR - DBS OR OODB OR OODBS OR RDB OR RDBS) ()TRIGGER?
S7	830	S1(10N)S2
S8	125	S2(10N)S3
S9	76	(S7 OR S8) (10N) (S4 OR S5)
S10	0	S3(5N)S6
S11	7	(S7 OR S8) (S)S4(S)S5
S12	44	(S7 OR S8) (5N) (S4 OR S5)
S13	50	S11 OR S12
S14	28	RD (unique items)
S15	28	S14 NOT PY>2002
S16	28	S15 NOT PD>20020221
File 275:		Gale Group Computer DB(TM) 1983-2004/Jun 03 (c) 2004 The Gale Group
File 47:		Gale Group Magazine DB(TM) 1959-2004/Jun 01 (c) 2004 The Gale group
File 75:		TGG Management Contents(R) 86-2004/May W4 (c) 2004 The Gale Group
File 636:		Gale Group Newsletter DB(TM) 1987-2004/Jun 02 (c) 2004 The Gale Group
File 16:		Gale Group PROMT(R) 1990-2004/Jun 03 (c) 2004 The Gale Group
File 624:		McGraw-Hill Publications 1985-2004/Jun 03 (c) 2004 McGraw-Hill Co. Inc
File 484:		Periodical Abs Plustext 1986-2004/May W5 (c) 2004 ProQuest
File 613:		PR Newswire 1999-2004/Jun 03 (c) 2004 PR Newswire Association Inc
File 813:		PR Newswire 1987-1999/Apr 30 (c) 1999 PR Newswire Association Inc
File 141:		Readers Guide 1983-2004/May (c) 2004 The HW Wilson Co
File 239:		Mathsci 1940-2004/Jul (c) 2004 American Mathematical Society
File 553:		Wilson Bus. Abs. FullText 1982-2004/May (c) 2004 The HW Wilson Co
File 621:		Gale Group New Prod. Annou. (R) 1985-2004/Jun 01 (c) 2004 The Gale Group
File 674:		Computer News Fulltext 1989-2004/May W3 (c) 2004 IDG Communications
File 88:		Gale Group Business A.R.T.S. 1976-2004/Jun 02 (c) 2004 The Gale Group
File 635:		Business Dateline(R) 1985-2004/Jun 02 (c) 2004 ProQuest Info&Learning
File 15:		ABI/Inform(R) 1971-2004/Jun 02 (c) 2004 ProQuest Info&Learning
File 9:		Business & Industry(R) Jul/1994-2004/Jun 02 (c) 2004 The Gale Group
File 13:		BAMP 2004/May W2 (c) 2004 The Gale Group
File 810:		Business Wire 1986-1999/Feb 28 (c) 1999 Business Wire
File 610:		Business Wire 1999-2004/Jun 02 (c) 2004 Business Wire.
File 647:		CMP Computer Fulltext 1988-2004/May W4 (c) 2004 CMP Media, LLC
File 148:		Gale Group Trade & Industry DB 1976-2004/Jun 02 (c) 2004 The Gale Group

16/3,K/5 (Item 5 from file: 275)  
DIALOG(R)File 275:Gale Group Computer DB(TM)  
(c) 2004 The Gale Group. All rts. reserv.

01669633 SUPPLIER NUMBER: 15032001 (USE FORMAT 7 OR 9 FOR FULL TEXT)  
The ASK Group launches enhanced DB2 gateway. (OpenIngres/Enterprise Access  
for DB2 gateway software)  
Pallatto, John  
PC Week, v11, n8, p51(2)  
Feb 28, 1994  
ISSN: 0740-1604 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT  
WORD COUNT: 386 LINE COUNT: 00030

... software is required on the mainframe, Donovan said.  
The new gateway also will work with **precompiled** COBOL database  
**procedures** using static SQL queries stored **within** DB2, he said. It also  
will work with ad hoc SQL queries.  
Replication services are...

16/3,K/7 (Item 7 from file: 275)  
DIALOG(R)File 275:Gale Group Computer DB(TM)  
(c) 2004 The Gale Group. All rts. reserv.

01591646 SUPPLIER NUMBER: 13667551 (USE FORMAT 7 OR 9 FOR FULL TEXT)  
Raima Database Server: a veteran software company enters a new age ready to  
compete. (Raima Corp.'s SQL client/server database management system)  
(The Server Side) (Column)

Roti, Steve

DBMS, v6, n4, p83(2)

April, 1993

DOCUMENT TYPE: Column ISSN: 1041-5173 LANGUAGE: ENGLISH

RECORD TYPE: FULLTEXT; ABSTRACT

WORD COUNT: 2085 LINE COUNT: 00163

... server features such as stored procedures, user-defined functions, triggers, and event alerters. RDS stored **procedures** are **precompiled** groups of SQL statements stored in the server's system catalog for speedy execution at **run time**. They can accept input arguments and return results from multiple SELECT statements. The limitation to...

...they can only contain SQL statements, they cannot perform any conditional logic and control flow **within** the procedure. All SQL statements in a stored procedure execute sequentially unless a **runtime** error occurs. In the case of an error, the procedure terminates. Here is a simple...

16/3,K/12 (Item 12 from file: 275)  
DIALOG(R)File 275:Gale Group Computer DB(TM)  
(c) 2004 The Gale Group. All rts. reserv.

01314935 SUPPLIER NUMBER: 07864548 (USE FORMAT 7 OR 9 FOR FULL TEXT)  
Easy API, paltry documentation. (Database Services component of IBM's OS/2  
Extended Edition 1.1 Database Manager) (evaluation)  
Mirecki, Ted; Erickson, Michelle  
PC Week, v6, n44, p101(1)  
Nov 6, 1989  
DOCUMENT TYPE: evaluation ISSN: 0740-1604 LANGUAGE: ENGLISH  
RECORD TYPE: FULLTEXT; ABSTRACT  
WORD COUNT: 1328 LINE COUNT: 00108

... word:  
EXEC SQL  
select name, salary from personnel;  
The program is first processed by a **precompiler** that converts such  
statements to comments and replaces them with **procedure** calls to server  
**run - time** routines. The output of the **precompiler** is a modified source  
program that is then processed by the normal compiler for the...

16/3,K/15 (Item 15 from file: 275)  
DIALOG(R)File 275:Gale Group Computer DB(TM)  
(c) 2004 The Gale Group. All rts. reserv.

01240476 SUPPLIER NUMBER: 06297206 (USE FORMAT 7 OR 9 FOR FULL TEXT)  
New on the market. (product announcement)  
Muchmore, Michael W.  
PC Magazine, v7, n8, p51(2)  
April 26, 1988  
DOCUMENT TYPE: product announcement ISSN: 0888-8507 LANGUAGE:  
ENGLISH RECORD TYPE: FULLTEXT  
WORD COUNT: 1239 LINE COUNT: 00097

... techniques, but 3-2-1 Blastoff improves calculation efficiency by  
translating 1-2-3 cell **formulas** into **machine code**. The compiled  
spreadsheets run only **inside** 1-2-3 and can be altered, although changes  
to existing formulas will slow down...

16/3,K/22 (Item 2 from file: 621)  
DIALOG(R) File 621:Gale Group New Prod.Annou.(R)  
(c) 2004 The Gale Group. All rts. reserv.

01037472 Supplier Number: 40015349 (USE FORMAT 7 FOR FULLTEXT)  
The release of FUTURE86, a next generation language, compiler and  
environment for IBM PC, AT and compatibles has been announced by  
Development Associates, a 17 year old Santa Ana, California electronics  
and software company.

PR Newswire, pN/A  
April, 1987  
Language: English Record Type: Fulltext  
Document Type: Newswire; Trade  
Word Count: 496

... is inherently rommable, reentrant and recursive.

Unique among existing languages, the programmer may freely mix  
assembly language and high level statements within procedures  
. This  
results in easy optimizing of applications, trading off speed and  
size in a controlled...



Set	Items	Description
S1	15764	(LOWLEVEL OR LOW()LEVEL OR ASSEMBLY OR MACHINE) (N) (LANGUAG- E? OR CODE?)
S2	5842545	EXPRESSION? OR PROCEDURE? OR FORMULA? OR ALGORITHM?
S3	969	PRECOMPILE? OR PRE()COMPILE?
S4	38916	RUNTIME? OR RUN()TIME?
S5	5370518	EMBED? OR INTEGRAL? OR WITHIN? OR INTERNAL? OR INSIDE? OR - INTEGRATE?
S6	107	(DATABASE OR DATA() (BANK? OR BASE?) OR DATABANK? OR DB OR - DBS OR OODB OR OODBS OR RDB OR RDBS) ()TRIGGER?
S7	2584	S1 AND S2
S8	314	S2 AND S3
S9	3	S7 AND S8
S10	33	(S7 OR S8) AND S4 AND S5
S11	1	S3 AND S6
S12	37	S9 OR S10 OR S11
S13	37	S12 NOT PY>2002
S14	26	RD (unique items)
File	8: Ei Compendex(R) 1970-2004/May W4	(c) 2004 Elsevier Eng. Info. Inc.
File	35: Dissertation Abs Online 1861-2004/May	(c) 2004 ProQuest Info&Learning
File	65: Inside Conferences 1993-2004/May W5	(c) 2004 BLDSC all rts. reserv.
File	2: INSPEC 1969-2004/May W4	(c) 2004 Institution of Electrical Engineers
File	94: JICST-EPlus 1985-2004/May W2	(c) 2004 Japan Science and Tech Corp(JST)
File	111: TGG Natl. Newspaper Index(SM) 1979-2004/Jun 03	(c) 2004 The Gale Group
File	233: Internet & Personal Comp. Abs. 1981-2003/Sep	(c) 2003 EBSCO Pub.
File	6: NTIS 1964-2004/May W5	(c) 2004 NTIS, Intl Cpyrght All Rights Res
File	144: Pascal 1973-2004/May W4	(c) 2004 INIST/CNRS
File	34: SciSearch(R) Cited Ref Sci 1990-2004/May W4	(c) 2004 Inst for Sci Info
File	99: Wilson Appl. Sci & Tech Abs 1983-2004/Apr	(c) 2004 The HW Wilson Co.

14/5/1 (Item 1 from file: 8)  
DIALOG(R)File 8: Ei Compendex(R)  
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

04912591 E.I. No: EIP98014024795

Title: **Inversion algorithm for digital simulation**

Author: Maurer, Peter M.

Corporate Source: Univ of South Florida, Tampa, FL, USA

Source: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems v 16 n 7 Jul 1997. p 762-769

Publication Year: 1997

CODEN: ITCSDI ISSN: 0278-0070

Language: English

Document Type: JA; (Journal Article) Treatment: A; (Applications); T; (Theoretical)

Journal Announcement: 9803W3

Abstract: The inversion **algorithm** is an event-driven **algorithm**, whose performance rivals or exceeds that of levelized compiled code simulation, even at activity rates of 50% or more. The inversion **algorithm** has several unique features, the most remarkable of which is the size of the **run - time** code. The basic **algorithm** can be implemented using no more than a page of **run - time** code, although in practice, it is more efficient to provide several different variations of the basic **algorithm**. The **run - time** code is independent of the circuit under test, so the **algorithm** can be implemented either as a compiled code or an interpreted simulator with little variation in performance. Because of the small size of the **run - time** code, the **run - time** portions of the inversion **algorithm** can be implemented in **assembly language** for peak efficiency, and still can be retargeted for new platforms with little effort. (Author abstract) 20 Refs.

Descriptors: **Integrated** circuit layout; Computer simulation; **Algorithms**; Codes (symbols); Computer aided design; **Integrated** circuit testing

Identifiers: Event driven **algorithm**; **Run time** code

Classification Codes:

714.2 (Semiconductor Devices & Integrated Circuits); 723.5 (Computer Applications); 723.2 (Data Processing)

714 (Electronic Components); 723 (Computer Software)

71 (ELECTRONICS & COMMUNICATIONS); 72 (COMPUTERS & DATA PROCESSING)

14/5/11 (Item 3 from file: 35)  
DIALOG(R) File 35:Dissertation Abs Online  
(c) 2004 ProQuest Info&Learning. All rts. reserv.

1067531 ORDER NO: AAD89-12874

**EFFICIENTLY COMBINING LOGICAL CONSTRAINTS WITH FUNCTIONS**

Author: BOTHNER, PER MAGNUS ALFRED  
Degree: PH.D.  
Year: 1989  
Corporate Source/Institution: STANFORD UNIVERSITY (0212)  
ADVISER: BRIAN K. REID  
Source: VOLUME 50/04-B OF DISSERTATION ABSTRACTS INTERNATIONAL.  
PAGE 1497. 190 PAGES  
Descriptors: COMPUTER SCIENCE  
Descriptor Codes: 0984

A declarative program specifies a set of logical constraints between a set of objects. The computer searches for the objects satisfying the constraints (i.e., the solution). An imperative program must specify a detailed **algorithm** for finding the solution.

Declarative programming can increase productivity (by making programs shorter and simpler). However, some problems discourage its widespread use. This thesis discusses how to make declarative programming more useful, using the new language Q.

A denotational description of pure Q defines a powerful specification language. A "complete" implementation (that can find concrete solutions for any constraint) is theoretically impossible. We must settle for concrete solutions in limited but useful cases. A technique called narrowing combines the power of Prolog-style logic programming (logic variables, unification, non-determinism) with the notational convenience (true **expression** evaluation) and first-class function values of functional programming. Additionally, Q supports more general constraints, including automatic solution of linear equations. A novel technique implements logical constraints by **embedding** logic variables (representing parameters) **inside** functionally dependent variables (representing results of functional constraints).

**Run - time** efficiency requires an optimizing compiler. One should pay for extra power only when it is used. One optimization replaces expensive logic variables and unification with cheap simple variables and assignment. It works for any function written in a "functional" style. Also, the support for backtracking (and in general continuations) is non-intrusive, using the standard C calling convention and stack, combined with some stack manipulation tricks when required.

Logic languages have poorly supported data abstraction and type definition. Types in Q are coercion functions, and hence first-class values, so a parametric type is just a higher-order function. A class is a type derived from a function that creates new records. Classes are used as the basis of a powerful module facility. Special methods that implement "application" make functions into true objects (that can contain **internal** state).

Declarative languages usually communicate poorly with other languages, which complicates calling low-level or previously-written code. The Q implementation strives for maximum compatibility with C. It uses C as a "portable **assembly language**," with standard calling conventions and data formats.

14/5/12 (Item 4 from file: 35)  
DIALOG(R)File 35:Dissertation Abs Online  
(c) 2004 ProQuest Info&Learning. All rts. reserv.

0985332 ORDER NO: AADDX-80955

**THE GENERATION OF CONCURRENT CODE FOR DECLARATIVE LANGUAGES (HYBRID)**

Author: ROTHWELL, NICHOLAS JOHN

Degree: PH.D

Year: 1986

Corporate Source/Institution: UNIVERSITY OF NEWCASTLE UPON TYNE (UNITED KINGDOM) (0682)

Source: VOLUME 49/02-B OF DISSERTATION ABSTRACTS INTERNATIONAL.  
PAGE 0469. 235 PAGES

Descriptors: COMPUTER SCIENCE

Descriptor Codes: 0984

Available from UMI in association with The British Library.

This thesis presents an approach to the implementation of declarative languages on a simple, general purpose concurrent architecture. The safe exploitation of the available concurrency is managed by relatively sophisticated code generation techniques to transform programs into an intermediate concurrent **machine code**. Compilation techniques are discussed for  $\text{\$cal F\$-HYBRID}$ , a strongly typed applicative language, and for  $\text{\$cal L\$-HYBRID}$ , a concurrent, nondeterministic logic language.

An approach is presented for  $\text{\$cal F\$-HYBRID}$  whereby the style of programming influences the concurrency utilised when a program executes. Code transformation techniques are presented which generalise tail-recursion optimisation, allowing many recursive functions to be modelled by perpetual processes. A scheme is also presented to allow parallelism to be increased by the use of local declarations, and constrained by the use of special forms of identity function.

In order to preserve determinism in the language, a novel fault handling mechanism is used, whereby exceptions generated at **run - time** are treated as a special class of values **within** the language.

A description is given of  $\text{\$cal L\$-HYBRID}$ , a dialect of the nondeterministic logic language Concurrent Prolog. The language is **embedded within** the applicative language  $\text{\$cal F\$-HYBRID}$ , yielding a combined applicative and logic programming language. Various cross-calling techniques are described, including the use of applicative scoping rules to allow local logical assertions.

A description is given of a polymorphic typechecking **algorithm** for logic programs, which allows different instances of clauses to unify objects of different types. The concept of a method is derived to allow unification information to be passed as an implicit argument to clauses which require it. In addition, the typechecking **algorithm** permits higher-order objects such as functions to be passed **within** arguments to clauses.

Using Concurrent Prolog's model of concurrency, techniques are described which permit compilation of  $\text{\$cal L\$-HYBRID}$  programs to abstract **machine code** derived from that used for the applicative language. The use of methods allows polymorphic logic programs to execute without the need for **run - time** type information in data structures.

14/5/14 (Item 6 from file: 35)  
DIALOG(R)File 35:Dissertation Abs Online  
(c) 2004 ProQuest Info&Learning. All rts. reserv.

908506 ORDER NO: AAD86-03164

**DESIGN OF PORTABLE DIRECT EXECUTING LANGUAGES FOR INTERACTIVE SIMULATION**

Author: VAKILZADIAN, HAMID

Degree: PH.D.

Year: 1985

Corporate Source/Institution: THE UNIVERSITY OF ARIZONA (0009)

Source: VOLUME 46/12-B OF DISSERTATION ABSTRACTS INTERNATIONAL.

PAGE 4321. 175 PAGES

Descriptors: COMPUTER SCIENCE

Descriptor Codes: 0984

DESIRE P is a general purpose continuous time simulation language suitable for interactive simulation, dynamic system study, mathematical modeling, process control analysis. It includes an interactive editor, file manipulation facilities, and graphic packages, making it a completely self-contained system. The PDP-11 version of DESIRE P handles 20 state variables, while the VAX/VMS version runs 150 or more. An interpreted job-control language serves for interactive program entry, editing and file operations, and for programming multirun simulation studies. The dynamic segment, containing differential equations in first-order form, is entered just like the job-control statements and accesses the same variables.

DESIRE P is largely written in PASCAL, and most of it can be transferred to different computers, with little change. The PASCAL implementation proves that the high-level language can be used to program direct executing languages, still keeping efficiency and speed comparable to **assembly language**. The runtime compiler of DESIRE P generates fast and efficient code. DESIRE P can incorporate existing and new **precompiled** FORTRAN numerical integration **algorithms**.

Set	Items	Description
S1	5461	(LOWLEVEL OR LOW()LEVEL OR ASSEMBLY OR MACHINE) (N) (LANGUAG- E? OR CODE?)
S2	737462	EXPRESSION? OR PROCEDURE? OR FORMULA? OR ALGORITHM?
S3	637	PRECOMPILE? OR PRE()COMPILE?
S4	11894	RUNTIME? OR RUN()TIME?
S5	1230711	EMBED? OR INTEGRAL? OR WITHIN? OR INTERNAL? OR INSIDE? OR - INTEGRATE?
S6	114	(DATABASE OR DATA() (BANK? OR BASE?) OR DATABANK? OR DB OR - DBS OR OODB OR OODBS OR RDB OR RDBS) ()TRIGGER?
S7	263	S1(10N)S2
S8	29	S2(10N)S3
S9	18	(S7 OR S8) (10N) (S4 OR S5)
S10	0	S3(S)S6
S11	2	S3 AND S6
S12	1	S5(S)S6(S)S1
S13	21	S9 OR S11 OR S12
S14	21	IDPAT (sorted in duplicate/non-duplicate order)
S15	20	IDPAT (primary/non-duplicate records only)
File 348:EUROPEAN PATENTS 1978-2004/May W04 (c) 2004 European Patent Office		
File 349:PCT FULLTEXT 1979-2002/UB=20040527,UT=20040520 (c) 2004 WIPO/Univentio		

15/3,K/5 (Item 5 from file: 348)  
DIALOG(R) File 348:EUROPEAN PATENTS  
(c) 2004 European Patent Office. All rts. reserv.

00597960

Programmable computer with automatic translation between source and object  
code with version control

Programmierbarer Rechner mit automatischer Übersetzung zwischen Quell - und  
Zielcode mit Versionüberwachung

Ordinateur programmable avec traduction automatique entre code source et  
code-cible avec controle de version

PATENT ASSIGNEE:

AMDAHL CORPORATION, (628802), 1250 East Arques Avenue, Sunnyvale, CA  
94088, (US), (applicant designated states:  
AT;BE;CH;DE;DK;ES;FR;GB;IT;LI;LU;NL;SE)

INVENTOR:

Knudsen, Helge, 46 Eighth Concession Road, RR#1, Freelon, Ontario L0R 1K0,  
(CA)

Chong, Daniel T., 2 Frankie Lane, Woodbridge, Ontario L4L 7J6, (CA)

Yaffe, John, 1239 Tenth Concession Road West, RR#3, Puslinch, Ontario N0B  
2J0, (CA)

Taughner, James, 1504 Chalice Crescent, Mississauga, Ontario L5C 1S3, (CA)

Robertson, Michael, 2641 Burnford Trail, Mississauga, Ontario L5M 4E1, (CA)

Plazak, Zbigniew, 70 Glen Agar, Etobicoke, Ontario M9B 5M1, (CA)

LEGAL REPRESENTATIVE:

Crawford, Andrew Birkby et al (29761), A.A. THORNTON & CO. Northumberland  
House 303-306 High Holborn, London WC1V 7LE, (GB)

PATENT (CC, No, Kind, Date): EP 588446 A2 940323 (Basic)  
EP 588446 A3 951115  
EP 588446 B1 990707

APPLICATION (CC, No, Date): EP 93203242 900904;

PRIORITY (CC, No, Date): US 402862 890901; US 450298 891213

DESIGNATED STATES: AT; BE; CH; DE; DK; ES; FR; GB; IT; LI; LU; NL; SE

RELATED PARENT NUMBER(S) - PN (AN):

EP 489861 (EP 909143406)

INTERNATIONAL PATENT CLASS: G06F-009/44; G06F-009/45; G06F-017/30;

ABSTRACT WORD COUNT: 594

LANGUAGE (Publication, Procedural, Application): English; English; English  
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	9927	1636
CLAIMS B	(German)	9927	1403
CLAIMS B	(French)	9927	1808
SPEC B	(English)	9927	16336
Total word count - document A			0
Total word count - document B			21183
Total word count - documents A + B			21183

...SPECIFICATION table name, it can also be generated from this table.

The detranslator converts the virtual **machine code** to tokenized  
representation of the rule language. The **algorithm** structure is written  
in the rules **within** the object-oriented operating system. It converts  
the post-fix form of the object level...

15/3,K/9 (Item 9 from file: 348)  
DIALOG(R)File 348:EUROPEAN PATENTS  
(c) 2004 European Patent Office. All rts. reserv.

00306058

Digital data processing system.

Digitales Datenverarbeitungssystem.

Système de traitement de données numériques.

PATENT ASSIGNEE:

DATA GENERAL CORPORATION, (410940), Route 9, Westboro Massachusetts 01581  
, (US), (applicant designated states: AT;BE;CH;DE;FR;GB;IT;LI;LU;NL;SE)

INVENTOR:

Bachman, Brett L., 214 W. Canton Street Suite 4, Boston Massachusetts  
02116, (US)  
Bernstein, David H., 41 Bay Colony Drive, Ashland Massachusetts 01721,  
(US)  
Bratt, Richard Glenn, 9 Brook Trail Road, Wayland Massachusetts 01778,  
(US)  
Clancy, Gerald F., 13069 Jaccaranda Center, Saratoga California 95070,  
(US)  
Gavrin, Edward S., Beaver Pond Road RFD 4, Lincoln Massachusetts 01773,  
(US)  
Gruner, Ronald Hans, 112 Dublin Wood Drive, Cary North Carolina 27514,  
(US)  
Jones, Thomas M. Jones, 300 Reade Road, Chapel Hill North Carolina 27514,  
(US)  
Katz, Lawrence H., 10943 S. Forest Ridge Road, Oregon City Oregon 97045,  
(US)  
Mundie, Craig James, 136 Castlewood Drive, Cary North Carolina, (US)  
Pilat, John F., 1308 Ravenhurst Drive, Raleigh North Carolina 27609, (US)  
Richmond, Michael S., Fearringtn Post Box 51, Pittsboro North Carolina  
27312, (US)  
Schleimer Stephen I., 1208 Ellen Place, Chapel Hill North Carolina 27514,  
(US)  
Wallach, Steven J., 12436 Green Meadow Lane, Saratoga California 95070,  
(US)  
Wallach, Walter, A., Jr., 1336 Medfield Road, Raleigh North Carolina  
27607, (US)

LEGAL REPRESENTATIVE:

Robson, Aidan John et al (69471), Reddie & Grose 16 Theobalds Road,  
London WC1X 8PL, (GB)

PATENT (CC, No, Kind, Date): EP 290111 A2 881109 (Basic)  
EP 290111 A3 890503  
EP 290111 B1 931222

APPLICATION (CC, No, Date): EP 88200917 820521;

PRIORITY (CC, No, Date): US 266404 810522

DESIGNATED STATES: AT; BE; CH; DE; FR; GB; IT; LI; LU; NL; SE

RELATED PARENT NUMBER(S) -PN (AN):

EP 67556 (EP 823025960)

INTERNATIONAL PATENT CLASS: G06F-009/30;

ABSTRACT WORD COUNT: 123

LANGUAGE (Publication,Procedural,Application): English; English; English

FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	EPBBF1	1044
CLAIMS B	(German)	EPBBF1	890
CLAIMS B	(French)	EPBBF1	1185
SPEC B	(English)	EPBBF1	154314
Total word count - document A			0
Total word count - document B			157433
Total word count - documents A + B			157433

...SPECIFICATION Names allow user's programs to be expressed in very compact code. Fewer SOPs than machine language instructions are required to express a user's program. Also, use of SOPs allows easier and simpler construction of compilers, and...or COBOL. User Language Instructions 402 are converted into a greater number of more detailed



- Machine Language Instructions 404, used within a machine to execute user's programs. Within the machine, Machine Language Instructions 404 are interpreted and executed by Microcode Instructions...level interrupts or calls pertaining to execution of a user's program may be stacked within MIS 424. Information stored in MIS 424 stack frames is generally information from GR 508...

...programs available in common to many users. In effect, a Procedure 602 contains the instructions ( **procedures** ) and data of a user's program.  
A Process 610 includes, as described above, a Macro-Stack (MAS) 502...

15/3,K/15 (Item 15 from file: 349)  
DIALOG(R)File 349:PCT FULLTEXT  
(c) 2004 WIPO/Univentio. All rts. reserv.

00577727 \*\*Image available\*\*

SYSTEM AND METHOD FOR RECURSIVE PATH ANALYSIS OF DBMS PROCEDURES  
SYSTEME ET PROCEDE D'ANALYSE RECURSIVE DE PROCEDURES DE SYSTEMES DE GESTION  
DE BASE DE DONNEES (SGBD)

Patent Applicant/Assignee:  
COMPUTER ASSOCIATES THINK INC,

Inventor(s):  
VINCENT John K,  
CHERNY Igor,

Patent and Priority Information (Country, Number, Date):

Patent: WO 200041100 A1 20000713 (WO 0041100)  
Application: WO 2000US276 20000106 (PCT/WO US00000276)  
Priority Application: US 99226939 19990108

Designated States: AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK  
EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS  
LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR  
TT UA UG UZ VN YU ZA ZW GH GM KE LS MW SD SL SZ TZ UG ZW AM AZ BY KG KZ  
MD RU TJ TM AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE BF BJ  
CF CG CI CM GA GN GW ML MR NE SN TD TG

Publication Language: English

Fulltext Word Count: 7796

Fulltext Availability:

Detailed Description  
Claims

Detailed Description

... and to use it for compiling and  
debugging code. One such attempt, involved binding together **pre -  
compiled**  
subroutines to form a complete host procedure object code, as described  
in US Patent No...4 and 5 together illustrates a flowchart showing the  
steps used to incorporate dependencies on **database triggers** and their  
dependencies into the dependency graph.

Figure 6 illustrates a flowchart showing the steps...the dependencies of  
Package and Type Specifications. The method also takes into  
considerations dependencies on **database Triggers**. Triggers are code  
objects that are executed automatically as a result of executing a Data  
...next database table dependency starting with the first one. Component  
44 gets the dependencies on **database triggers** of the database table.  
(Triggers are defined and described in Chapter 8, OracleTrm Server  
Application developers Guide, which is hereby incorporated fully herein  
by reference). Component 45 selects the next **database trigger**  
dependency starting with the first one. Component 46 parses the source  
code of the code...

Claim

... presentation  
tool.

17) A method of generating dependency information including  
dependencies of code objects on **database triggers**, the method  
comprising the steps of  
1) using a recursive algorithm for querying a database...

15/3,K/18 (Item 18 from file: 349)  
DIALOG(R)File 349:PCT FULLTEXT  
(c) 2004 WIPO/Univentio. All rts. reserv.

00446054 \*\*Image available\*\*  
OPENBUS SYSTEM FOR CONTROL AUTOMATION NETWORKS INCORPORATING FUZZY LOGIC  
CONTROL  
SYSTEME DE BUS OUVERT POUR RESEAUX D'AUTOMATISATION DE COMMANDE A COMMANDE  
LOGIQUE FLOUE

Patent Applicant/Assignee:

AZARYA Arnon,  
AZARYA Yitzhak,

Inventor(s):

AZARYA Arnon,  
AZARYA Yitzhak,

Patent and Priority Information (Country, Number, Date):

Patent: WO 9836518 A2 19980820

Application: WO 98IL43 19980129 (PCT/WO IL9800043)

Priority Application: US 97790974 19970130

Designated States: AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES  
FI GB GE GH GM GW HU ID IL IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD  
MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG US  
UZ VN YU ZW GH GM KE LS MW SD SZ UG ZW AM AZ BY KG KZ MD RU TJ TM AT BE  
CH DE DK ES FI FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN ML  
MR NE SN TD TG

Publication Language: English

Fulltext Word Count: 21643

Fulltext Availability:

Detailed Description

Detailed Description

... be described in more detail. As described previously, the fuzzy  
compiler functions to read the **formulas** or equations making up the  
membership function and generate **machine code** which can be executed  
on the moth coprocessor of the **embedded** processor.

During operation of the control application, when a **formula** or function  
must be calculated, control passes to the **machine code** associated  
with the **formula**. The **machine code** generated by the compiler may  
correspond to any or all of the following.

mathematical formulas...

Set	Items	Description
S1	1579	(LOWLEVEL OR LOW()LEVEL OR ASSEMBLY OR MACHINE) (N) (LANGUAG- E? OR CODE?)
S2	1240849	EXPRESSION? OR PROCEDURE? OR FORMULA? OR ALGORITHM?
S3	109	PRECOMPILE? OR PRE()COMPILE?
S4	2597	RUNTIME? OR RUN()TIME?
S5	3232160	EMBED? OR INTEGRAL? OR WITHIN? OR INTERNAL? OR INSIDE? OR - INTEGRATE?
S6	10	(DATABASE OR DATA() (BANK? OR BASE?) OR DATABANK? OR DB OR - DBS OR OODB OR OODBS OR RDB OR RDBS) ()TRIGGER?
S7	156	S1 AND S2
S8	12	S2 AND S3
S9	33	(S7 OR S8) AND (S4 OR S5)
S10	43	S9 OR S6
S11	39	S10 AND IC=G06F?
S12	39	IDPAT (sorted in duplicate/non-duplicate order)
S13	38	IDPAT (primary/non-duplicate records only)
File 347:JAPIO Nov 1976-2004/Jan(Updated 040506)		
(c) 2004 JPO & JAPIO		
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200434		
(c) 2004 Thomson Derwent		

13/5/4 (Item 4 from file: 350)  
DIALOG(R) File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

015396709 \*\*Image available\*\*  
WPI Acc No: 2003-458849/200344  
XRPX Acc No: N03-364871

Operation of a computer system on which a control program runs that calls  
at least one COM (common object module) in a manner that does not require  
compilation of the control program

Patent Assignee: SIEMENS AG (SIEI )

Inventor: HELM M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
DE 10141799	A1	20030320	DE 1041799	A	20010827	200344 B

Priority Applications (No Type Date): DE 1041799 A 20010827

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
DE 10141799	A1		9	G06F-009/40	

Abstract (Basic): DE 10141799 A1

NOVELTY - Method for operation of a computer system on which a  
control program runs that calls at least one COM (common object module)  
with a **procedure** and **procedure** interface via a custom interface to  
the COM. Accordingly a data record corresponding to a the **procedure**  
interface of a **procedure** is written to a buffer memory, the data  
record is copied to the stack and the **procedure** is called when the  
control program branches to the **procedure** start address.

USE - Computer operation method for running source code compiled  
into **machine code**, where such code contains subroutines or  
**procedures** with a COM object with a custom type interface.

ADVANTAGE - The inventive method allows access to a custom type  
interface without requiring compilation of the control program, whereas  
previously the routine had to be compiled at **run time** when  
parameters were defined. The method can also be used for automatic  
testing of COM components.

DESCRIPTION OF DRAWING(S) - Figure shows a flow diagram of an  
inventive method.

reading in of script file (100)  
reading in of type information (105)  
calling of subroutine. (120)  
pp; 9 DwgNo 4/4

Title Terms: OPERATE; COMPUTER; SYSTEM; CONTROL; PROGRAM; RUN; CALL; ONE;  
COMMON; OBJECT; MODULE; MANNER; REQUIRE; COMPILE; CONTROL; PROGRAM

Derwent Class: T01

International Patent Class (Main): G06F-009/40

File Segment: EPI

13/5/6 (Item 6 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

014938990 \*\*Image available\*\*  
WPI Acc No: 2002-759699/200282  
XRPX Acc No: N02-598237

Database trigger implementing method for relational database  
management system, involves inserting data reference corresponding to  
associated column in transition table, for performing data access during  
trigger operation

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC )  
Inventor: CHEN Y C S; CORNWELL K L; DANG T; MOORE B R; SHYAM K  
Number of Countries: 001 Number of Patents: 001  
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20020138497	A1	20020926	US 2001817501	A	20010326	200282 B

Priority Applications (No Type Date): US 2001817501 A 20010326

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 20020138497	A1		9	G06F-007/00	

Abstract (Basic): US 20020138497 A1

NOVELTY - A row in base tables (10a,10b) related to a detected  
trigger event, is determined. A data reference is generated for each  
column related to the trigger event. The data reference is inserted in  
the transition table column, using which data access is performed  
during trigger operation.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the  
following:

- (1) System for implementing database trigger ; and
- (2) Program for implementing database trigger .

USE - For implementing database trigger for maintaining data  
records of relational database management system (RDBMS).

ADVANTAGE - Reference to data such as large object data (LOB) is  
stored in the transition table instead of actual data, thus reduces  
space in the transition table.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of  
the database trigger implementing system.

Base tables (10a,10b)  
pp; 9 DwgNo 1/3

Title Terms: DATABASE; TRIGGER; IMPLEMENT; METHOD; RELATED; DATABASE;  
MANAGEMENT; SYSTEM; INSERT; DATA; REFERENCE; CORRESPOND; ASSOCIATE;  
COLUMN; TRANSITION; TABLE; PERFORMANCE; DATA; ACCESS; TRIGGER; OPERATE

Derwent Class: T01

International Patent Class (Main): G06F-007/00

File Segment: EPI

13/5/9 (Item 9 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

014736069 \*\*Image available\*\*

WPI Acc No: 2002-556773/200259

Related WPI Acc No: 2002-130241; 2002-154991; 2002-216327; 2002-329207;  
2002-519064

XRFX Acc No: N02-440690

Software program optimizing method in DSP, involves developing optimized  
form of software that are dependent on target processor and include  
portions coded in high and low level languages of target processor

Patent Assignee: CADENCE DESIGN SYSTEMS INC (CADE-N)

Inventor: CANUT F; DERRAS M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20020066088	A1	20020530	US 2000216746	A	20000703	200259 B
			US 2001765916	A	20010118	

Priority Applications (No Type Date): US 2000216746 P 20000703; US  
2001765916 A 20010118

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
US 20020066088	A1	16	G06F-009/45	Provisional application US 2000216746

Abstract (Basic): US 20020066088 A1

NOVELTY - A software program is optimized such that a resulting  
optimized form of software is independent of a target processor and is  
coded in a high level language. The optimized form of the software  
program is further optimized such that the resulting form of the  
software are dependent on the target processor and include portions  
coded in high and low level languages of the target processor.

USE - For optimizing software program for target processor e.g.  
digital signal processor (DSP), microcontroller with hard wired  
circuitry such as application-specific integrated circuit (ASIC),  
field programmable gate array (FPGA) or other logic devices.

ADVANTAGE - Provides optimized software program for target  
processor in order to meet specific performance objectives. Avoids  
overhead introduced by recursive calls, moving loop invariant  
expressions out of the loops, and reducing the scope of the variables.

DESCRIPTION OF DRAWING(S) - The figure shows a flow diagram of the  
optimizing software program for target processor.

pp; 16 DwgNo 1/8

Title Terms: SOFTWARE; PROGRAM; OPTIMUM; METHOD; DEVELOP; OPTIMUM; FORM;  
SOFTWARE; DEPEND; TARGET; PROCESSOR; PORTION; CODE; HIGH; LOW; LEVEL;  
LANGUAGE; TARGET; PROCESSOR

Derwent Class: T01

International Patent Class (Main): G06F-009/45

File Segment: EPI

13/5/15 (Item 15 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

013408697 \*\*Image available\*\*  
WPI Acc No: 2000-580635/200055  
XRPX Acc No: N00-429782

Trigger gateway operating method for trigger processing in active  
databases, in which trigger gateway receives database commands destined  
for database system, and processes triggers associated with database  
commands

Patent Assignee: LUCENT TECHNOLOGIES INC (LUCE ); AVAYA TECHNOLOGY CORP  
(AVAY-N)

Inventor: ARLEIN R M; LIEUWEN D F; MICHAEL G C; ORDILLE J J; SILVA J F  
Number of Countries: 028 Number of Patents: 004

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1022663	A2	20000726	EP 2000300173	A	20000111	200055 B
CA 2293933	A1	20000722	CA 2293933	A	20000105	200055
JP 2000215092	A	20000804	JP 20007950	A	20000117	200055
US 6594656	B1	20030715	US 99235730	A	19990122	200348

Priority Applications (No Type Date): US 99235730 A 19990122

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 1022663	A2	E	13	G06F-017/30	
Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT					
LI LT LU LV MC MK NL PT RO SE SI					
CA 2293933	A1	E			
JP 2000215092	A		12	G06F-012/00	
US 6594656	B1			G06F-017/30	

Abstract (Basic): EP 1022663 A2

NOVELTY - The trigger gateway operating method involves receiving a  
database command destined for a database management system, identifying  
at least one trigger associated with the database command, and  
processing the at least one trigger.

DETAILED DESCRIPTION - The trigger gateway operating method is  
implemented in active **database trigger** processing and involves  
processing using a trigger gateway. The trigger gateway is located at a  
communication point between a user and a database system, and receives  
database commands destined for the database and processes triggers  
associated with database commands. Where appropriate, the trigger  
gateway forwards the database command to the database. A trigger  
server, which is located remote from the trigger gateway may execute  
trigger actions in response to trigger execution requests sent from the  
trigger gateway. INDEPENDENT CLAIMS are included for; a **database**  
**trigger** gateway, which is independent of a database management system.

USE - Operating trigger gateway which is independent of a database  
management system, for processing **database triggers** each of which  
has an associated action.

ADVANTAGE - Enables active **database trigger** processing using a  
highway trigger.

DESCRIPTION OF DRAWING(S) - The drawing shows a diagram of a system  
in which the invention may be implemented.

Database (102)  
Trigger gateway (104)  
Network (106)  
User (108)  
Trigger action server (110)  
pp; 13 DwgNo 1/7

Title Terms: TRIGGER; GATEWAY; OPERATE; METHOD; TRIGGER; PROCESS; ACTIVE;  
TRIGGER; GATEWAY; RECEIVE; DATABASE; COMMAND; DATABASE; SYSTEM; PROCESS;  
TRIGGER; ASSOCIATE; DATABASE; COMMAND

Derwent Class: T01

International Patent Class (Main): G06F-012/00 ; G06F-017/30

File Segment: EPI



13/5/18 (Item 18 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

012867036 \*\*Image available\*\*  
WPI Acc No: 2000-038869/200003  
XRPX Acc No: N00-029299

**Distributed code conversion system for program compiling**  
Patent Assignee: SUN MICROSYSTEMS INC (SUNM )  
Inventor: CARTWRIGHT R S  
Number of Countries: 075 Number of Patents: 003  
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 9957635	A1	19991111	WO 99US8938	A	19990426	200003 B
AU 9937596	A	19991123	AU 9937596	A	19990426	200016
US 6075942	A	20000613	US 9872308	A	19980504	200035

Priority Applications (No Type Date): US 9872308 A 19980504

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9957635	A1	E	56	G06F-009/45	
Designated States (National): AL AU BA BB BG BR CA CN CU CZ EE GD GE HR HU ID IL IN IS JP KP KR LC LK LR LT LV MG MK MN MX NO NZ PL RO SG SI SK SL TR TT UA UZ VN YU					
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW NL OA PT SD SE SL SZ UG ZW					
AU 9937596	A			G06F-009/45	Based on patent WO 9957635
US 6075942	A			G06F-009/445	

Abstract (Basic): WO 9957635 A1

NOVELTY - Source code is translated into register oriented **machine code** which is then translated into stack oriented **machine code**. The resultant stack oriented **machine code** is transmitted from one computer system to other computer system including microprocessor. The computer system translates stack oriented **machine code** into register oriented **machine code** based on which microprocessor is operated.

DETAILED DESCRIPTION - The source code is translated into register oriented **machine code** that consists of sequence of **machine code** instruction which comprises machine operation code and one register code. The machine operation code represents arithmetic or logical operation and register code explicitly specifies machine register from which operation is to draw its operand and store its results. The stack oriented machine operation code explicitly specifies the stack oriented **machine code** local variable associated with machine register. The variable code specifies loading the operand stack with local variable. The register oriented **machine code** translated by one computer system is same as the register oriented **machine code** translated by other computer system. An INDEPENDENT CLAIM is also included for distributed code conversion method.

USE - For variable program compilers.

ADVANTAGE - Executes **machine code** in optimizing manner by assigning registers in accordance with predetermined association of registers with local variables. By concurrent performance with **run - time** computer system's compiling, delay that optimization **procedure** imposes compromises benefits of optimizing global register allocation.

DESCRIPTION OF DRAWING(S) - The figure shows flow chart illustrating distributed code conversion system.

pp; 56 DwgNo 4/10

Title Terms: DISTRIBUTE; CODE; CONVERT; SYSTEM; PROGRAM; COMPILE  
Derwent Class: T01  
International Patent Class (Main): G06F-009/445 ; G06F-009/45  
File Segment: EPI

13/5/20 (Item 20 from file: 350)  
DIALOG(R) File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

012628465 \*\*Image available\*\*  
WPI Acc No: 1999-434569/199937  
XRPX Acc No: N99-323898

Distributed system used in computer networks e.g. internet - has  
arithmetic processing unit which assigns each partial process to  
corresponding processing client computer, and transmits virtual machine  
code of each partial process to processing client computers

Patent Assignee: TOSHIBA KK (TOKE )

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 11175485	A	19990702	JP 97346369	A	19971216	199937 B

Priority Applications (No Type Date): JP 97346369 A 19971216

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 11175485	A		8	G06F-015/16	

Abstract (Basic): JP 11175485 A

NOVELTY - The arithmetic processing unit assigns each partial process to the corresponding processing client computer (20), and transmits the virtual machine code corresponding to each partial process to each processing client computers. DETAILED DESCRIPTION - A server (10) has a juxtaposition calculating service unit (14) that informs a calculation dividing unit (13) of the contents of a particular process if a processing demand is received from the exclusive application unit (32) of a process-demanding client computer (30). The dividing unit divides the contents of that process into corresponding partial processes, and requests that praxis from an arithmetic processing unit (12). An INDEPENDENT CLAIM is also included for a juxtaposition calculation control procedure .

USE - For computer networks e.g. internet.

ADVANTAGE - Simplifies system configuration, thus negating use of special hardware. Newly-utilized network unit can be recognized dynamically. Effectively increases processing speed within environment in which high and low-speed machines are intermingled.  
DESCRIPTION OF DRAWING(S) - The figure shows the component diagram of the client-server system. (10) Server; (12) Arithmetic processing unit; (13) Calculation dividing unit; (14) Juxtaposition calculating service unit; (20) Processing client computer; (30) Process-demanding client computer; (32) Exclusive application unit.

Dwg.1/3

Title Terms: DISTRIBUTE; SYSTEM; COMPUTER; NETWORK; ARITHMETIC; PROCESS;  
UNIT; ASSIGN; PROCESS; CORRESPOND; PROCESS; CLIENT; COMPUTER; TRANSMIT;  
VIRTUAL; MACHINE; CODE; PROCESS; PROCESS; CLIENT; COMPUTER

Derwent Class: T01

International Patent Class (Main): G06F-015/16

International Patent Class (Additional): G06F-013/00

File Segment: EPI

13/5/21 (Item 21 from file: 350)  
DIALOG(R) File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

012217841 \*\*Image available\*\*  
WPI Acc No: 1999-023947/199902  
XRPX Acc No: N99-018445

**Source code processing system - converts source code to virtual machine code containing compiled instructions representing expressions and uncompiled instructions representing pre- runtime executable structure constructs**

Patent Assignee: SYBASE INC (SYBA-N)  
Inventor: GUILLEN J; LEASK J M  
Number of Countries: 001 Number of Patents: 001  
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5838980	A	19981117	US 94183480	A	19940118	199902 B
			US 97882247	A	19970625	

Priority Applications (No Type Date): US 94183480 A 19940118; US 97882247 A 19970625

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
US 5838980	A	18	G06F-009/45	Cont of application US 94183480

Abstract (Basic): US 5838980 A

The processor includes a language sub system which determines the syntactic and semantic scheme according to which source code (12) is prepared for compilation. The source code is parsed and tokenized to produce virtual **machine code** (16) including compiled instructions representing **expressions** and uncompiled instructions representing pre- **runtime** executable structure constructs by a compiler (14). A virtual machine processing system connected to compiler receives the virtual **machine code** and executes the pre- **runtime** executable structure constructs.

ADVANTAGE - Reduces quantity of **runtime** code by compiling source code to manageable object code blocks, thereby reducing execution time.

Dwg.1/10

Title Terms: SOURCE; CODE; PROCESS; SYSTEM; CONVERT; SOURCE; CODE; VIRTUAL; MACHINE; CODE; CONTAIN; COMPILE; INSTRUCTION; REPRESENT; EXPRESS; INSTRUCTION; REPRESENT; PRE; EXECUTE; STRUCTURE; CONSTRUCTION

Derwent Class: T01

International Patent Class (Main): G06F-009/45

File Segment: EPI

13/5/22 (Item 22 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

011562311 \*\*Image available\*\*  
WPI Acc No: 1997-538792/199750  
XRPX Acc No: N97-448399

Method of inserting assembly code routine into source code routine in  
data processor system compiler - involves scanning instructions and  
operands of template to determine whether all instructions and operands  
of template are included in set of recognised instructions

Patent Assignee: SUN MICROSYSTEMS INC (SUNM )

Inventor: GOEBEL K J

Number of Countries: 007 Number of Patents: 005

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 806725	A2	19971112	EP 97106998	A	19970428	199750 B
JP 10097430	A	19980414	JP 97130393	A	19970502	199825
US 5815719	A	19980929	US 96643895	A	19960507	199846
EP 806725	B1	20030827	EP 97106998	A	19970428	200358
DE 69724322	E	20031002	DE 624322	A	19970428	200372
			EP 97106998	A	19970428	

Priority Applications (No Type Date): US 96643895 A 19960507

Cited Patents: No-SR.Pub

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 806725	A2	E	14	G06F-009/45	
Designated States (Regional): DE FR GB NL SE					
JP 10097430	A		10	G06F-009/45	
US 5815719	A			G06F-009/45	
EP 806725	B1	E		G06F-009/45	
Designated States (Regional): DE FR GB NL SE					
DE 69724322	E			G06F-009/45	Based on patent EP 806725

Abstract (Basic): EP 806725 A

The method involves providing an **assembly code** template with the instructions and operands of the **assembly code** routine. A set of recognised instructions recognisable by the compiler for code optimisation is provided. The instructions and operands of the template are scanned to determine whether all instructions and operands of the template are included in the set of recognised instructions.

The **assembly code** is transformed into an intermediate form unable by the compiler. The source code is transformed into an intermediate form unable by the compiler. The intermediate form **assembly code** and the intermediate form source code are combined. Physical register assignments **within** the template are identified, and transformed into virtual register assignments.

ADVANTAGE - Permits early inlining of **assembly code** templates in appropriate cases so that **assembly code** can be subjected to all optimisation **procedures** incorporated into compiler. Provides more efficient **assembly code**.

Dwg.1/6b

Title Terms: METHOD; INSERT; ASSEMBLE; CODE; ROUTINE; SOURCE; CODE; ROUTINE  
; DATA; PROCESSOR; SYSTEM; COMPILE; SCAN; INSTRUCTION; OPERAND; TEMPLATE;  
DETERMINE; INSTRUCTION; OPERAND; TEMPLATE; SET; RECOGNISE; INSTRUCTION

Derwent Class: T01

International Patent Class (Main): G06F-009/45

File Segment: EPI

DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

011064668    \*\*Image available\*\*  
WPI Acc No: 1997-042593/199704  
XRPX Acc No: N97-035461

Run - time checking method for compiled programming development  
environments - involves executing run - time checking functions to  
determine if invalid operations aggregate data items and pointer and  
reporting errors to users

Patent Assignee: NAT INSTR CORP (NAIN-N)  
Inventor: BELLIN J; CRANK E  
Number of Countries: 001 Number of Patents: 001  
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5583988	A	19961210	US 94208559	A	19940309	199704 B

Priority Applications (No Type Date): US 94208559 A 19940309  
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5583988	A		62	G06F-011/00	

Abstract (Basic): US 5583988 A

The method involves compiling a source code file into executable object comprising of **machine language** instructions. The source code file includes aggregate data items and pointers and also includes **expressions** which manipulate the aggregate data items and pointers. The compilation involves creating data structures for aggregate data items and pointers in the source code file, and inserting calls to **run - time** checking functions for one or more **expressions** in the source code file which manipulate the aggregate data items and pointers.

The executable code is executed. The execution involves executing one or more of the **run - time** checking functions to determine if invalid operations occur in the **expression** which manipulate the aggregate data items and pointers. If an invalid operation is found to occur after the step of executing one or more of the **run - time** checking functions errors are reported to the user.

USE/ADVANTAGE - Detects errors are detected at precise moment that respective C language restriction is violated. User is provided with direct indication of problem and location in the source code file where violation occurred.

Dwg.2b/31b

Title Terms: RUN; TIME; CHECK; METHOD; COMPILE; PROGRAM; DEVELOP;  
ENVIRONMENT; EXECUTE; RUN; TIME; CHECK; FUNCTION; DETERMINE; INVALID;  
OPERATE; AGGREGATE; DATA; ITEM; POINT; REPORT; ERROR; USER

Derwent Class: T01

International Patent Class (Main): G06F-011/00

File Segment: EPI

13/5/29 (Item 29 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

008293872 \*\*Image available\*\*  
WPI Acc No: 1990-180873/199024  
XRPX Acc No: N90-140553

Database management system for translating program source code -  
containing a language independent portion and a communicating language  
dependent portion

Patent Assignee: IBM CORP (IBMC ); INT BUSINESS MACHINES CORP (IBMC )  
Inventor: CHANG P Y T; COYLE D J; HARGORVE D D C; HIDALGO D S; CHANG P Y;  
HARGROVE D C

Number of Countries: 005 Number of Patents: 003

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 373132	A	19900613	EP 89850418	A	19891127	199024 B
BR 8906000	A	19900619				199029
US 5230049	A	19930720	US 88277367	A	19881129	199330

Priority Applications (No Type Date): US 88277367 A 19881129

Cited Patents: 4.Jnl.Ref; A3...9147; NoSR.Pub

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 373132	A				

Designated States (Regional): DE FR GB

US 5230049 A 7 G06F-009/45

Abstract (Basic): EP 373132 A

The language specific translator (12) selects supplemental language statements from an input file (10), containing many source code statements, and converts them into a language independent format. These are then passed to the independent translator.

The independent portion translates the language independent statements into a list of tasles to be performed, identifying sequences of calls to library **procedures** which must be made in order to implement the communicated statements. These tasks are communicated back to the language dependent portion in a language independent format, which translates them into **procedure** calls in the host language.

ADVANTAGE - This invention simplifies the writing of **pre - compilers** and allows them to be written without detailed knowledge of an underlying supplemental system. (2pp Dwg.No.1/2)

Title Terms: DATABASE; MANAGEMENT; SYSTEM; TRANSLATION; PROGRAM; SOURCE;  
CODE; CONTAIN; LANGUAGE; INDEPENDENT; PORTION; COMMUNICATE; LANGUAGE;  
DEPEND; PORTION

Derwent Class: T01

International Patent Class (Main): G06F-009/45

International Patent Class (Additional): G06F-015/40

File Segment: EPI

13/5/31 (Item 31 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

001754504

WPI Acc No: 1977-K1009Y/197745

Programming language expression processor - uses processor language  
resembling high-level language to accelerate translation process

Patent Assignee: CONTROL PROBLEM INS (CONT-R)

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
SU 519715	A	19760726				197745 B

Priority Applications (No Type Date): SU 1995907 A 19740212

Abstract (Basic): SU 519715 A

Programming language **expression** processor is designed for use as a specialised data processing device **within** a multi-processor computing system. The proposed model is aimed at simplifying and accelerating the process of translation from high-level languages into **internal** computer language. It differs from other similar machines by containing a control symbol memory stack (7), a control symbol analyser (4), a control symbol counter (5), as well as address and operand forming unit (2), index position counter (16) and a file number register (17). The new elements render the **internal machine language** considerably more sophisticated.

The similarity between the high-level programming language and the corresponding programme written in the processor's **internal** language makes programme completion control during checkout much simpler.

Title Terms: PROGRAM; LANGUAGE; EXPRESS; PROCESSOR; PROCESSOR; LANGUAGE;  
RESEMBLE; HIGH; LEVEL; LANGUAGE; ACCELERATE; TRANSLATION; PROCESS

Derwent Class: T01

International Patent Class (Additional): G06F-015/04

File Segment: EPI

13/5/33 (Item 33 from file: 347)  
DIALOG(R)File 347:JAPIO  
(c) 2004 JPO & JAPIO. All rts. reserv.

06762001 \*\*Image available\*\*  
METHOD AND DEVICE FOR PROCESSING EXCEPTION AS REGULAR CONTROL FLOW

PUB. NO.: 2000-347872 [JP 2000347872 A]  
PUBLISHED: December 15, 2000 (20001215)  
INVENTOR(s): CLICK JR CLIFFORD N  
VICK CHRISTOPHER A  
PALECZNY MICHAEL H  
APPLICANT(s): SUN MICROSYST INC  
APPL. NO.: 2000-122769 [JP 2000122769]  
FILED: April 24, 2000 (20000424)  
PRIORITY: 298354 [US 99298354], US (United States of America), April  
23, 1999 (19990423)  
INTL CLASS: G06F-009/45

#### ABSTRACT

PROBLEM TO BE SOLVED: To process an exception as part of the flow of regular program control by preparing the final **internal expression** of a source code by excluding a call related to a partial exception examination and generating a **machine code** related to the final **internal expression**.

SOLUTION: An optimizer 222 is constituted so as to exclude an excessive 'goto' command as part of regular processing. Besides, the optimizer 222 optimizes an intermediate **internal expression** 226 as well so as to exclude an unused code such as a dead code, for example, from a final **internal expression** 230. The final **internal expression** 230 is used by the optimizer 222 for preparing a **machine code** 242. A call existent in the intermediate **internal expression** 226 is excluded from the final **internal expression** 230, but the final **internal expression** 230 can be handled as an **expression** arithmetically corresponding to the intermediate **internal expression** 226, on the other hand. By excluding the call of a function for throwing down the exception, an unused exception object is excluded.

COPYRIGHT: (C)2000,JPO



13/5/38 (Item 38 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2004 JPO & JAPIO. All rts. reserv.

01312748 \*\*Image available\*\*  
COMPILATION PROCESSING SYSTEM

PUB. NO.: 59-024348 [JP 59024348 A]  
PUBLISHED: February 08, 1984 (19840208)  
INVENTOR(s): NAKAMURA YOSHIHIRO  
HASU SHIGEHARU  
APPLICANT(s): FUJITSU LTD [000522] (A Japanese Company or Corporation), JP  
(Japan)  
APPL. NO.: 57-134984 [JP 82134984]  
FILED: July 31, 1982 (19820731)  
INTL CLASS: [3] G06F-009/44  
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)  
JOURNAL: Section: P, Section No. 277, Vol. 08, No. 117, Pg. 96, May  
31, 1984 (19840531)

#### ABSTRACT

PURPOSE: To enable to refer a variable, i.e., a subtracting number of **procedure** with an **assembly language** described within the **procedure**, by processing the part described in the **assembly language** during the compilation.

CONSTITUTION: A sentence structure analysis part 27 and a meaning analysis part 28 of an **assembly language** processing part 26 check both the sentence structure and the meaning of a statement using the **assembly language** and then calls out a subtracting number deciding part 29. The part 29 decides whether the variable used in the statement of **assembly language** is equal to the subtracting number of a **procedure** on the basis of the information of a working table 31. When the variable is equal to the subtracting number, an alteration additional processing part 30 is started. The part 30 has the mnemonic alteration and addition so that the variable of the subtracting number of the **procedure** corresponds to an actual memory region at the part where the **assembly language** is written with reference to a symbol table 32 and an **internal** style produced by a sentence structure analysis part 24 for high-class languages and a meaning analysis part 25.

Set	Items	Description
S1	4	AU=(STEGELMANN, R? OR STEGELMANN R?)
S2	6	AU=(CHAWARE J? OR CHAWARE, J?)
S3	0	S1 AND S2
S4	10	(S1 OR S2) AND IC=G06F?
S5	10	IDPAT (sorted in duplicate/non-duplicate order)
S6	6	IDPAT (primary/non-duplicate records only)

File 347:JAPIO Nov 1976-2004/Jan(Updated 040506)  
(c) 2004 JPO & JAPIO

File 348:EUROPEAN PATENTS 1978-2004/May W04  
(c) 2004 European Patent Office

File 349:PCT FULLTEXT 1979-2002/UB=20040527,UT=20040520  
(c) 2004 WIPO/Univentio

File 350:Derwent WPIX 1963-2004/UD,UM &UP=200434  
(c) 2004 Thomson Derwent

6/5/1 (Item 1 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

015382054 \*\*Image available\*\*  
WPI Acc No: 2003-442996/200342  
XRPX Acc No: N03-353621

Index selection method for relational database system, involves  
generating candidate indexes from workload received in index wizard  
client module, and optimizing candidate indexes to generate the  
recommended index

Patent Assignee: NCR INT INC (NATC ); BROWN D P (BROW-I); CHAWARE J  
(CHAW-I); KOPPURAVURI M (KOPP-I)

Inventor: BROWN D P; CHAWARE J ; KOPPURAVURI M

Number of Countries: 031 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1302870	A2	20030416	EP 2002256677	A	20020925	200342 B
US 20030093408	A1	20030515	US 2001977038	A	20011012	200342

Priority Applications (No Type Date): US 2001977038 A 20011012

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 1302870	A2	E	68	G06F-017/30	

Designated States (Regional): AL AT BE BG CH CY CZ DE DK EE ES FI FR GB  
GR IE IT LI LT LU LV MC MK NL PT RO SE SI SK TR

US 20030093408 A1 G06F-007/00

Abstract (Basic): EP 1302870 A2

NOVELTY - A workload containing a set of queries comprising a  
structured query language (SQL) statements of the database system (14),  
is received in an index wizard client module (24) to generate a set of  
candidate indexes. The indexes are eliminated based on one or more  
predetermined criteria such as change rate. An optimizer is invoked to  
generate the recommended index from the set of candidate indexes.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the  
following:

- (1) database system; and
- (2) Stored software.

USE - For selecting recommended index for database management  
systems such as relational database management systems (RDBMS).

ADVANTAGE - Achieves proper selection of suitable indexes easily  
and effectively and thereby obtains optimal database performance.  
Reduces load on optimizer as it is provided with comparatively lesser  
number of candidate indexes after criteria based elimination, rather  
than all possible candidate indexes.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of  
network of target database systems, a client system and a test system.

test system (10)  
data network (12)  
database systems (14)  
client system (20)  
index wizard client module (24)  
pp; 68 DwgNo 1/47

Title Terms: INDEX; SELECT; METHOD; RELATED; DATABASE; SYSTEM; GENERATE;  
CANDIDATE; INDEX; RECEIVE; INDEX; CLIENT; MODULE; OPTIMUM; CANDIDATE;  
INDEX; GENERATE; RECOMMENDED; INDEX

Derwent Class: T01

International Patent Class (Main): G06F-007/00 ; G06F-017/30

File Segment: EPI

6/5/2 (Item 2 from file: 350)  
DIALOG(R)File 350:Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

015371470 \*\*Image available\*\*  
WPI Acc No: 2003-432408/200341

XRPX Acc No: N03-345176

**Demographics information presentation method in relational database management system, involves displaying demographics information in graphical format based on user-selected menu item in graphical user interface screen**

Patent Assignee: NCR INT INC (NATC ); BROWN D P (BROW-I); CHAWARE J (CHAW-I)

Inventor: BROWN D P; CHAWARE J

Number of Countries: 031 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1302886	A1	20030416	EP 2002256687	A	20020925	200341 B
US 20030088546	A1	20030508	US 2001976632	A	20011012	200341

Priority Applications (No Type Date): US 2001976632 A 20011012

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
EP 1302886	A1	E	32 G06F-017/60	

Designated States (Regional): AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI SK TR

US 20030088546 A1 G06F-017/30

Abstract (Basic): EP 1302886 A1

NOVELTY - A graphical user interface screen is provided in a display (26) of a client system (20). The demographics information is displayed in a graphical format or text format in response to selection of a menu item by the user in the graphical user interface screen.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

- (1) database system; and
- (2) demographics information presentation program.

USE - For presenting demographics information in multi-node parallel processing system such as massively parallel processing (MPP) system, database system (claimed) such as relational database management system (RDBMS).

ADVANTAGE - As various types of demographic information needed for performing tests and analysis to determine the performance of queries in a target database system are easily collected on a per-access module basis and presented in graphical or text format, the performance of the database systems is greatly improved.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of the network of systems, including target database system, client system and test system.

client system (20)

display (26)

pp; 32 DwgNo 1/20

Title Terms: INFORMATION; PRESENT; METHOD; RELATED; DATABASE; MANAGEMENT; SYSTEM; DISPLAY; INFORMATION; GRAPHICAL; FORMAT; BASED; USER; SELECT; MENU; ITEM; GRAPHICAL; USER; INTERFACE; SCREEN

Derwent Class: T01

International Patent Class (Main): G06F-017/30 ; G06F-017/60

International Patent Class (Additional): G06F-007/00

File Segment: EPI

6/5/3 (Item 3 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

015371469 \*\*Image available\*\*

WPI Acc No: 2003-432407/200341

XRPX Acc No: N03-345175

**Relational database management system operation method involves collecting statistics regarding one attribute of table in system, based on request containing specified percentage of rows to be read**

Patent Assignee: NCR INT INC (NATC ); BROWN D P (BROW-I); CHAWARE J (CHAW-I)

Inventor: BROWN D P; CHAWARE J

Number of Countries: 031 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1302871	A2	20030416	EP 2002256686	A	20020925	200341 B
US 20030088579	A1	20030508	US 2001976634	A	20011012	200341

Priority Applications (No Type Date): US 2001976634 A 20011012

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 1302871	A2	E	20	G06F-017/30	
Designated States (Regional): AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI SK TR					
US 20030088579	A1			G06F-017/00	

Abstract (Basic): EP 1302871 A2

NOVELTY - The method involves collecting statistics regarding at least one attribute of a table in the database system, based on scanning a sample of rows of the table, with samples being less than the rows. The statistics collection is performed, when a request containing the percentage of rows to be read is transmitted to the system.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

- (1) database management system; and
- (2) computer program for operating database system.

USE - For statistics collection in relational database management systems (RDBMS).

ADVANTAGE - Enables faster collection of statistics in the database system, and thereby improves the operation of certain components such as optimizer and other tools in the system.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of a query capture database.

pp; 20 DwgNo 3/12

Title Terms: RELATED; DATABASE; MANAGEMENT; SYSTEM; OPERATE; METHOD;  
COLLECT; STATISTICAL; ONE; ATTRIBUTE; TABLE; SYSTEM; BASED; REQUEST;  
CONTAIN; SPECIFIED; PERCENTAGE; ROW; READ

Derwent Class: T01

International Patent Class (Main): G06F-017/00 ; G06F-017/30

File Segment: EPI

6/5/4 (Item 4 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

015136261 \*\*Image available\*\*

WPI Acc No: 2003-196787/200319

XRFX Acc No: N03-156089

Subject table accessing method in relational database management system,  
involves selecting records such that one action involved in request,  
access partition of subject table

Patent Assignee: NCR CORP (NATC )

Inventor: CHEN J; HOANG C K; HODGENS M A; KAUFMANN F S; STEGELMANN R G E

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 6470331	B1	20021022	US 99454730	A	19991204	200319 B

Priority Applications (No Type Date): US 99454730 A 19991204

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6470331	B1		8	G06F-017/30	

Abstract (Basic): US 6470331 B1

NOVELTY - A subject table (300) is partitioned across processing units and a request to access the table is split into messages comprising actions assigned to the processing units. The records of the actions are retrieved from the partition of the subject table and

stored into a spool table (304). The records are selected such that action involved in the request access the partition of the subject table.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the following:

- (1) Subject table accessing apparatus; and
  - (2) Article of manufacture comprising recorded medium storing subject table accessing program.
- USE - For accessing subject table in relational database management system of computers e.g. main frame computer, PC, etc.
- ADVANTAGE - The performance of the database partitions is improved effectively.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of the data structure.

Subject table (300)

Spool table (304)

pp; 8 DwgNo 3/4

Title Terms: SUBJECT; TABLE; ACCESS; METHOD; RELATED; DATABASE; MANAGEMENT; SYSTEM; SELECT; RECORD; ONE; ACTION; REQUEST; ACCESS; PARTITION; SUBJECT; TABLE

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

6/5/5 (Item 5 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

014928526 \*\*Image available\*\*

WPI Acc No: 2002-749235/200281

XRPX Acc No: N02-589950

Modifications replication method for RDBMS, involves mapping processor of primary system to new processor of subscriber system, and mapping its sequence number to new sequence number of new processor

Patent Assignee: NCR CORP (NATC )

Inventor: STEGELMANN R G E

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 6438558	B1	20020820	US 99471736	A	19991223	200281 B

Priority Applications (No Type Date): US 99471736 A 19991223

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6438558	B1	16	G06F-017/30		

Abstract (Basic): US 6438558 B1

NOVELTY - The change row message (214) identifies the processor in the primary system and includes a sequence number for the processor in the primary system. The identified processor is remapped to the new processor in the subscriber system. The included sequence number is mapped to a new sequence number for the new processor in the subscriber system.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the following:

- (1) Apparatus for replicating modifications; and
  - (2) Article of manufacture for replication modifications.
- USE - Used in relational database management system (RDBMS) for replicating modifications.

ADVANTAGE - Optimizes the database access on parallel processing computer systems and improves the performance of database partitions managed by a parallel processing computer system. The modifications are applied in a correct order on the subscriber system.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of a modification replication system.

Change row message (214)

pp; 16 DwgNo 2/4

Title Terms: MODIFIED; REPLICAS; METHOD; MAP; PROCESSOR; PRIMARY; SYSTEM;  
NEW; PROCESSOR; SUBSCRIBER; SYSTEM; MAP; SEQUENCE; NUMBER; NEW; SEQUENCE;  
NUMBER; NEW; PROCESSOR  
Derwent Class: T01  
International Patent Class (Main): G06F-017/30  
File Segment: EPI

6/5/6 (Item 6 from file: 350)  
DIALOG(R) File 350: Derwent WPIX  
(c) 2004 Thomson Derwent. All rts. reserv.

014014006 \*\*Image available\*\*  
WPI Acc No: 2001-498220/200155  
XRPX Acc No: N01-369262

Subject table accessing method in parallel processing computer system,  
involves executing some step messages of split triggering event and  
action to be executed simultaneously and in parallel by processing units  
Patent Assignee: NCR INT INC (NATC ); NCR CORP (NATC )  
Inventor: CHEN J J; HOANG C K; HODGENS M A; KAUFMANN F S; STEGELMANN R G E  
; CHEN J

Number of Countries: 027 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1107135	A2	20010613	EP 2000309980	A	20001109	200155 B
US 6374236	B1	20020416	US 99454729	A	19991204	200232

Priority Applications (No Type Date): US 99454729 A 19991204

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 1107135	A2	E	15	G06F-017/30	
Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT					
LI LT LU LV MC MK NL PT RO SE SI TR					
US 6374236	B1			G06F-017/30	

Abstract (Basic): EP 1107135 A2

NOVELTY - Triggering event is split into step messages to access  
subject table partitioned across processing units of computer system.  
Triggered action performed in response to triggering event is split  
into step messages to access a spool table. Each step message of  
triggering event and triggered action is assigned to processing units  
that manage partitions of subject table and spool table and some step  
messages are executed simultaneously and in parallel by processing  
unit.

DETAILED DESCRIPTION - A trigger is defined for the subject table  
that is partitioned across processing units of computer system. The  
spool table generated in response to triggering event stores necessary  
records accessed from subject table by triggering event has partitions  
corresponding to partitions of subject table. INDEPENDENT CLAIMS are  
also included for the following:

(a) Computer program;

(b) Subject table accessing apparatus

USE - For optimal trigger based accessing method of table in  
database management system e.g. relational database management system  
(RDBMS) in parallel processing computer system.

ADVANTAGE - The triggering events and triggered actions are  
executed in parallel resulting in faster execution using fewer  
resources, thereby improving response time and throughput.

DESCRIPTION OF DRAWING(S) - The figure shows the flowchart  
illustrating the steps necessary for interpretation and execution of  
SQL statement.

pp; 15 DwgNo 2/4

Title Terms: SUBJECT; TABLE; ACCESS; METHOD; PARALLEL; PROCESS; COMPUTER;  
SYSTEM; EXECUTE; STEP; MESSAGE; SPLIT; TRIGGER; EVENT; ACTION; EXECUTE;  
SIMULTANEOUS; PARALLEL; PROCESS; UNIT

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

Set	Items	Description
S1	0	AU=(STEGELMANN, R? OR STEGELMANN R?)
S2	0	AU=(CHAWARE J? OR CHAWARE, J?)
File	2:INSPEC	1969-2004/May W4 (c) 2004 Institution of Electrical Engineers
File	6:NTIS	1964-2004/May W5 (c) 2004 NTIS, Intl Cpyrght All Rights Res
File	8:Ei Compendex(R)	1970-2004/May W4 (c) 2004 Elsevier Eng. Info. Inc.
File	34:SciSearch(R)	Cited Ref Sci 1990-2004/May W4 (c) 2004 Inst for Sci Info
File	35:Dissertation Abs Online	1861-2004/May (c) 2004 ProQuest Info&Learning
File	65:Inside Conferences	1993-2004/May W5 (c) 2004 BLDSC all rts. reserv.
File	148:Gale Group Trade & Industry DB	1976-2004/Jun 02 (c)2004 The Gale Group
File	94:JICST-EPlus	1985-2004/May W2 (c)2004 Japan Science and Tech Corp(JST)
File	275:Gale Group Computer DB(TM)	1983-2004/Jun 03 (c) 2004 The Gale Group
File	674:Computer News Fulltext	1989-2004/May W3 (c) 2004 IDG Communications
File	647:CMP Computer Fulltext	1988-2004/May W4 (c) 2004 CMP Media, LLC



Set	Items	Description
S1	140	(LOWLEVEL OR LOW()LEVEL OR ASSEMBLY OR MACHINE) (N) (LANGUAG- E? OR CODE?)
S2	4624	EXPRESSION? OR PROCEDURE? OR FORMULA? OR ALGORITHM?
S3	35	PRECOMPILE? OR PRE()COMPILE?
S4	1140	RUNTIME? OR RUN()TIME?
S5	25377	EMBED? OR INTEGRAL? OR WITHIN? OR INTERNAL? OR INSIDE? OR - INTEGRATE?
S6	10	(DATABASE OR DATA() (BANK? OR BASE?) OR DATABANK? OR DB OR - DBS OR OODB OR OODBS OR RDB OR RDBS) () TRIGGER?
S7	3	S1(4N)S2
S8	1	S2(4N)S3
S9	3	(S7 OR S8) AND (S4 OR S5)
S10	0	S3 AND S6
S11	3	S9 NOT PY>2002

File 256:SoftBase:Reviews,Companies&Prods. 82-2004/May  
(c)2004 Info.Sources Inc